

# Klasificiranje grafičkih prikaza korištenjem plitkih neuronskih mreža

---

**Hofer, Filip**

**Undergraduate thesis / Završni rad**

**2022**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:200:013019>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-12-23**

*Repository / Repozitorij:*

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I  
INFORMACIJSKIH TEHNOLOGIJA**

**Sveučilišni studij**

**KLASIFICIRANJE GRAFIČKIH PRIKAZA  
KORIŠTENJEM PLITKIH NEURONSKIH MREŽA**

**Završni rad**

**Filip Hofer**

**Osijek, 2022.**

# SADRŽAJ

1. UVOD .....	1
1.1. Zadatak završnog rada.....	1
2. PREGLED PODRUČJA TEME .....	2
2.1. Strojno učenje.....	2
2.1.1. Nadzirano strojno učenje.....	3
2.1.2. Nenadzirano strojno učenja.....	3
2.1.3. Podržano učenje .....	3
2.2 Neuronske mreže.....	4
2.2.1. Konvolucijske neuronske mreže .....	6
2.2.2. AlexNet .....	6
2.2.3. VGGnet .....	7
2.2.4. GoogLeNet.....	8
2.2.5. LeNet-5.....	8
2.3. Duboko učenje.....	9
2.3.1. Arhitektura dubokih neuronskih mreža.....	10
3. KLASIFICIRANJE GRAFIČKIH PRIKAZA KORIŠTENJEM PLITKIH NEURONSKIH MREŽA.....	12
3.1. Obrada skupa podataka .....	13
3.2. Treniranje mreža .....	13
3.3. Plitka neuronska mreža .....	13
3.4. Duboka neuronska mreža .....	15
3.5. Rezultati zadatka .....	18
4. ZAKLJUČAK .....	28
LITERATURA.....	29
SAŽETAK.....	30
ABSTRACT.....	31
ŽIVOTOPIS .....	32

# 1. UVOD

Postoje mnogi kognitivni zadaci koji su laki za ljude, ali izuzetno teški za računala. Umjetna inteligencija razvila je sustave za rješavanje takvih problema. Tradicionalni pristupi umjetne inteligencije temelje se na ideji simboličkog predstavljanja znanja i zaključivanja kao simboličkih operacija. U današnje vrijeme umjetna inteligencija postala je dio svake industrije koja koristi računala i računalnu moć kao resurs. Iako su svi dosadašnji oblici umjetne inteligencije ograničeni na rješavanje samo određenih problema i nemaju vlastitu svijest za razumijevanje. Te ograničene funkcije dovoljne su kako bi se takav oblik tehnologije koristio za automatiziranje mnoštva procesa. Umjetna inteligencija kao grana znanosti, koja se tek sada počela ozbiljnije razvijati, pušta svoje korijene već više od 50 godina. Napredak u računalnoj moći dao joj je veliki vjetar u leđa te zbog toga tek sada vidimo nekakve ozbiljne pomake u tom području. To nam omogućuje da obučavanje neuronske mreže svedemo na jednostavniji zadatak za koji je potrebno samo nekoliko minuta rada na običnom prijenosnom ili stolnom računalu. Takva su poboljšanja privukla sve više i više inženjera da počnu eksperimentirati i istraživati ovo zanimljivo područje, što je rezultiralo mnogim novim bibliotekama i okvirima za različite programske jezike, uvelike ubrzavajući i olakšavajući korištenje algoritama strojnog učenja. Općenito govoreći, strojno učenje uključuje adaptivne mehanizme koji omogućuju računalima da skupljaju iskustvo, tj. uče kroz primjere i analogije. Učenje pruža mogućnosti za poboljšanje učinka pametnih sustava kroz vrijeme. Mehanizmi strojnog učenja temelj su adaptivnih sustava. Najpopularnije metode strojnog učenja su umjetne neuronske mreže i genetski algoritmi.

## 1.1. Zadatak završnog rada

Zadatak ovog završnog rada je proučiti i opisati plitke neuronske mreže te usporediti njihovu uspješnost u problemu klasificiranja različitih vrsta grafičkih prikaza. Potrebno je pronaći najmanju moguću konvolucijsku neuronsku mrežu sa što manjim brojem slojeva i usporediti je s nekom od najčešće korištenih neuronskih mreža kao što su VGG, ResNet, Inception, AlexNet i dr. Za usporedbu je potrebno koristiti jedinstveni podatkovni skup, npr. ICDAR 2019 dataset of chart images.

## 2. PREGLED PODRUČJA TEME

Ljudi su kroz povijest imali svakakve ideje na temu umjetne inteligencije koje su se pojavljivale u literaturi pa čak i u mitovima koji datiraju već od 16. stoljeća. Mit o Golemu najpoznatiji je zapis takve vrste, a radi se o magičnom biću stvorenom od gline, blata i kamena čija je svrha bila slušanje i izvršavanje naredbi te kako vrijeme teče on postaje sve snažniji. [1] Možemo povući analogiju na umjetnu inteligenciju jer umjetni sustavi rade bolje što više iskustva imaju.

Umjetna inteligencija je dio znanosti koja se bavi razvojem računala odnosno njihovih sposobnosti obavljanja zadataka za koje je potrebna određena inteligencija kako bi se izvršili. Te sposobnosti omogućuju računalima da se snalaze u novim situacijama koje do sada nisu prošli, da mogu učiti nove koncepte i donositi zaključke na temelju stečenog znanja. Umjetni inteligentni sustavi funkcioniraju na principu sličnom kao i ljudi. Ponašanje im je prilagodljivo, uče na temelju stečenog iskustva, velika količina znanja pohranjena u memoriju, imaju sposobnost komunikacije s čovjekom putem prirodnog jezika i govora. Iako je ljudska inteligencija sposobna obavljati razne funkcije pa čak i više funkcija odjednom, inteligentni sustavi su obično specijalizirani za obavljanje jedne radnje.

Začetnik umjetne inteligencije je John McCarthy. On je 1956. godine organizirao kongres na Dartmouth Collegeu na kojemu je i skovan izraz „umjetna inteligencija“. McCarthy je autor osnovnog programskog jezika umjetne inteligencije zvanog LISP. Već 1961. godine razvijen je šahovski program koji je bio na razini igre šahovskih majstora. [2] 1970 ih, nakon razvoja programskog jezika PROLOG nastao je i prvi zaista upotrebljivi sustav koji je koristio umjetnu inteligenciju a to je bio MYCIN. Sposobnosti tog sustava su da dijagnosticira bakterijske krvne infekcije te preporuči liječenje. Taj sustav je u nekim dijagnozama bio točniji od liječnika no zbog slabih računalnih performansi znanje mu je bilo preusko.

### 2.1. Strojno učenje

Svjedoci smo revolucije podataka, svake godine ih ima sve više i više. Sve više stvari se mjeri i povećava se broj atributa. Podatci više nisu samo u obliku nekakvih brojeva nego se tu pojavljuju slike, zvukovi, grafovi, web stranice itd. Svaka interakcija na internetu kao nuspojavu ima generiranje podataka. Senzori su po svuda, pametni satovi i pametni telefoni. Internet of things je također jedan od razloga popularnosti strojnog učenja.

Algoritmi strojnog učenja imaju zadatak pronaći uzorke i poveznice u podacima koje su relevantne za željeni ishod te na temelju toga odlučiti ili predviđati. Stroj se trenira tako što koristi algoritme i velike količine podataka na temelju kojih uči kako provesti zadatak. Istrenirani algoritmi svakodnevno se koriste za donošenje važnih odluka u medicinskoj dijagnostici, trgovanju, predviđanju potrošnje energije itd. Mnogi web portali koriste algoritme strojnog učenja kako bi analizirali navike svojih klijenata te na temelju toga plasirale podatke koje smatraju da klijent želi vidjeti. [3]

### **2.1.1. Nadzirano strojno učenje**

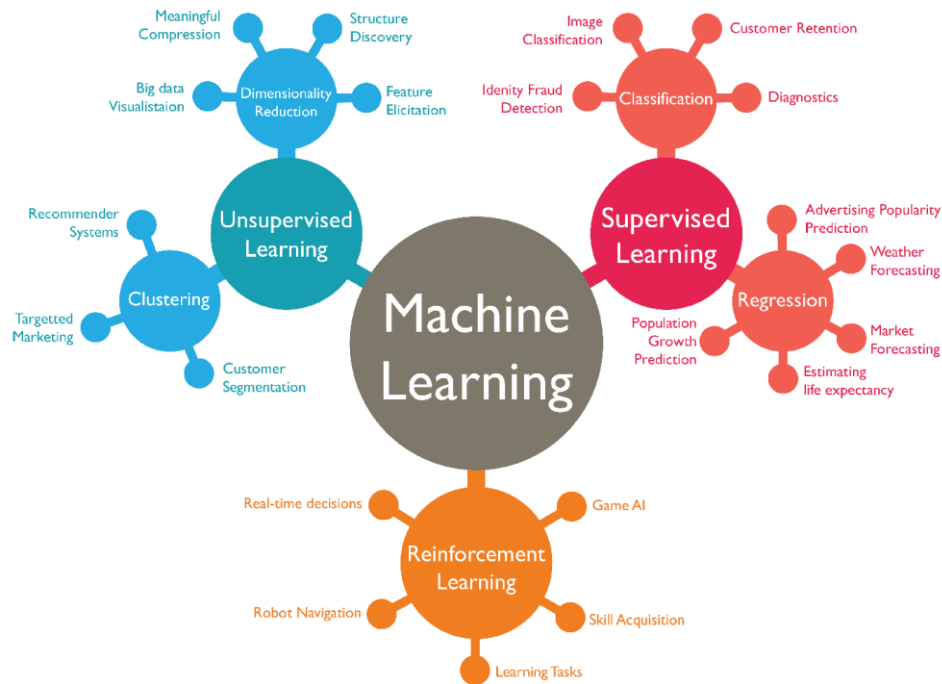
Cilj nadziranog strojnog učenja je odrediti nepoznatu funkcionalnu ovisnost između zadanih ulaznih veličina i željene izlazne veličine na temelju nekakvih podatkovnih primjera. Koristi se za predviđanje vrijednosti izlaznih veličina na temelju modela i ulaznih veličina. Model po kojem se vrši predviđanje obično se tretira kao crna kutija i nije od velike važnosti njegov oblik, bitno je samo koliko su mu dobre sposobnosti predviđanja. Također se modeli nadziranog strojnog učenja mogu koristiti i za zaključivanje. U tom slučaju pokušava se pronaći koje su ulazne veličine u vezi s izlaznom veličinom ili kakav je oblik funkcionalne povezanosti između svake ulazne veličine i izlaznih veličina.

### **2.1.2. Nenadzirano strojno učenja**

Kod ove vrste strojnog učenja na raspolaganju su samo podaci o ulaznim veličinama dok izlazna veličina ne postoji već je cilj modela da pronađe određene pravilnosti u podacima. Produkt takve analize može biti grupiranje podataka ili klaster analiza, smanjivanje dimenzionalnosti podataka i procjena gustoće vrijednosti. Ovom metodom mogu se riješiti problemi kao što su segmentacija kupaca na temelju nekakvih anketa, analiza društvenih mreža s ciljem plasiranja reklama te grupiranje nekakvih dokumenata po sličnostima.

### **2.1.3. Podržano učenje**

Kod podržanog učenja postoji akter koji pokušava samostalno otkriti optimalno ponašanje metodom pokušaja i pogrešaka. Akter poduzima korake u okruženju u kojem se našao kako bi riješio određen problem. Za svaku akciju akter dobiva povratnu informaciju u ovisnosti je li ta akcija dobra ili nije. Nakon određenog broja pokušaja akter bi trebao sam naučiti koji je najbolji način rješavanja problema u kojem se nalazi. Podržano učenje je iterativni postupak gdje se ponavlja jedna te ista stvar dok se ne dovede do zadovoljavajućeg ishoda. Primjenu ovog načina strojnog učenja možemo naći u robotici gdje se razni roboti snalaze u svom okolišu.

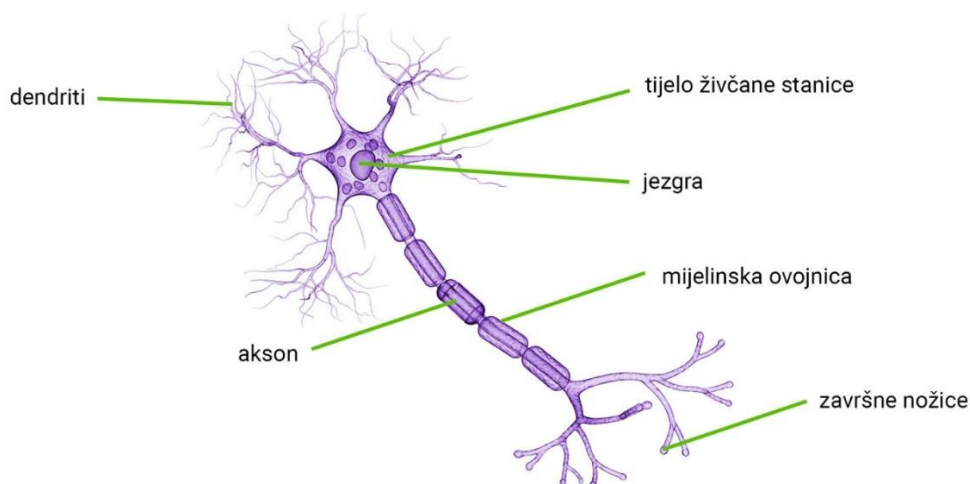


**Slika 2.1.** Vrste strojnog učenja<sup>1</sup>

## 2.2 Neuronske mreže

Jedan od najvećih misterija u medicini je mozak. Možemo reći da je ljudski mozak zapravo stroj jer izračunava ono što vidimo, čujemo i općenito osjećamo. Također, stvaramo misli i reagiramo na podražaje, kao što su govor ili pokret. Razumijevanje ljudskog mozga može nam pomoći da izgradimo bolja računala i strojeve da budu kao i mi, da razumiju okolinu i na taj način pomognu ljudskom rodu. [4] Inspiracija za koncept umjetne neuronske mreže nastala je od promatranja ljudskog mozga i njegovih funkcija. Neuron je osnovna jedinica za procesiranje u mozgu. Tijelo jednog neurona je okruženo dendritima koji služe za prijenos signala. Dendriti su veza između neurona odnosno jedan neuron je svojim dendritima vezan za velik broj drugih neurona. Svi ulazi jednog neurona ne utječu jednako na taj neuron odnosno njegovu reakciju. Neki su ulazi pobuđujući a neki su inhibitorni. Svaki neuron u sebi ima neku funkciju koja utječe na sumu svih ulaza tog neurona. Kada ta funkcija da odziv na pobudni signal tada se nekakav signal fizički propagira kroz akson tog neurona te se razgrana i šalje do sljedećih neurona. [5]

<sup>1</sup> <https://wordstream-files-prod.s3.amazonaws.com/s3fs-public/machine-learning.png>



**Slika 2.2.** Grafički prikaz neurona<sup>2</sup>

Perceptron je osnovni element umjetnih neuronskih mreža. On nam govori kako se nešto procesuirano odnosno opaža. Svaki ulaz u perceptron mora se pomnožiti s određenom težinom te se nakon toga svi ti umnošci sumiraju te se šalju na aktivacijsku funkciju. Ta funkcija služi kao prag za filtriranje podataka.

Umjetne neuronske mreže sastavljene su od niza jednostavnih elemenata tj. neurona koji su organizirani po slojevima i obrađuju podatke. Za svaki neuron postoji aktivacijska funkcija koja aktivira izlaz za određenu kombinaciju ulaza. Mogu se sastojati od više slojeva te u slojevima može biti veliki broj neurona. Slojeve dijelimo na ulazni sloj, skrivene slojeve i izlazni sloj. Ulazni sloj služi kako bi se primili podatci, skrivene slojevi su zaduženi za obradu podataka te izlazni sloj daje konačni rezultat. Cijela umjetna neuronska mreža je zapravo jedna velika funkcija koja za određene ulaze daje izlaz.

Neuronske mreže možemo razlikovati prema načinu spajanja pa tako postoje unaprijedne neuronske mreže, cikličke, jednoslojne ili višeslojne i djelomično ili potpuno povezane. Unaprijedne neuronske mreže su najjednostavnije jer tok podataka ide samo prema naprijed dok kod cikličkih mogu postojati povratne veze. Kada pričamo o broju slojeva, tada mreže mogu biti jednoslojne ili višeslojne te također ti slojevi ne moraju biti potpuno povezani već mogu biti i djelomično povezani.

<sup>2</sup> <https://edutorij.e-skole.hr/share/proxy/alfresco-noauth/edutorij/api/proxy-guest/4f77c550-2b26-4717-b65e-82b845685f3f/biologija-8/m03/kazalo-multimedije/index.html>



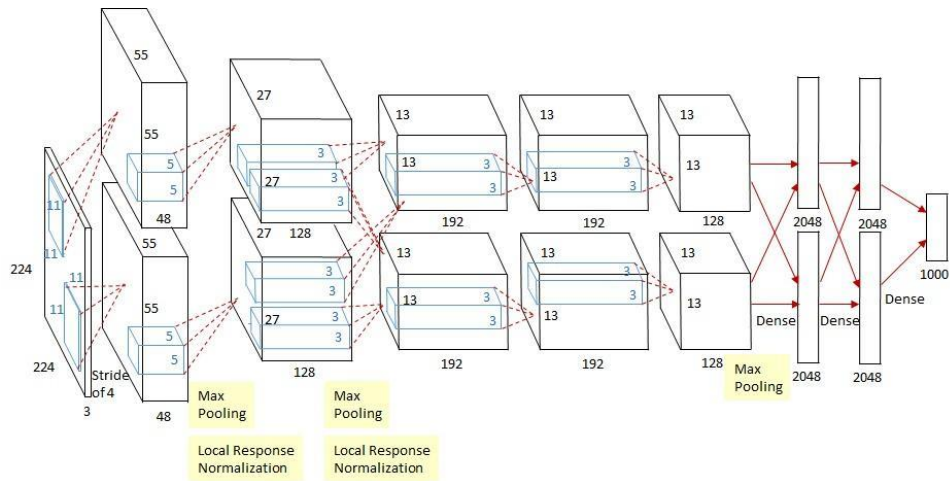
### **2.2.1. Konvolucijske neuronske mreže**

Konvolucijska neuronska mreža se razlikuje od obične neuronske mreže po tome što kao ulaz prima sliku. Samim time što je na ulazu slika imamo mogućnost programiranja određenih karakteristika u arhitekturu koje će učiniti unaprijednu funkciju učinkovitijom odnosno smanjiti će se broj parametara u mreži. Konvolucijske neuronske mreže su specifične po tome što imaju posložene neurone u trodimenzionalne strukture sa parametrima visine, širine i dubine. Visinu i širinu čine sama visina i širina slike dok dubinu čine kanali boje koje se nalaze na slici (zeleni, plava i crvena). Osnova konvolucijskih mreža je sami konvolucijski sloj čijom se primjenom u arhitekturi neuronske mreže zadržava prostorna struktura ulaznih podataka.[6]

Arhitektura konvolucijskih neuronskih mreža se programira kako bi bila dobra u raspoznavanju uzoraka i obrazaca. To mogu biti razni objekti, rubovi objekata, oblici i teksture. Za prepoznavanje obrazaca koriste se filtri koji pomažu lakšem prepoznavanju detalja kao što su ljudsko lice, lišće na granama i sl. Filtri su u obliku matrice sa kvadratnim dimenzijama koja se pomiče po slici u određenim koracima te tako detektira objekte na slici. Filtar pokriva malo prostorno područje no prostire se kroz cijelu dubinu slike odnosno kroz sva tri ulazna kanala slike. Produkt primjene filtra je aktivacijska mapa. To je dvodimenzionalna matrica koja sadrži odzive filtra na pojedinim dijelovima slike. Kod dizajniranja arhitekture konvolucijske neuronske mreže moguće je primijeniti više filtara što rezultira sa više aktivacijskih mapa koje se slažu jedna pored druge. Slojevi koji se mogu naći u arhitekturi konvolucijske neuronske mreže su ulazni sloj, konvolucijski sloj, sloj sažimanja, potpuno povezani sloj i sloj aktivacijske funkcije.

### **2.2.2. AlexNet**

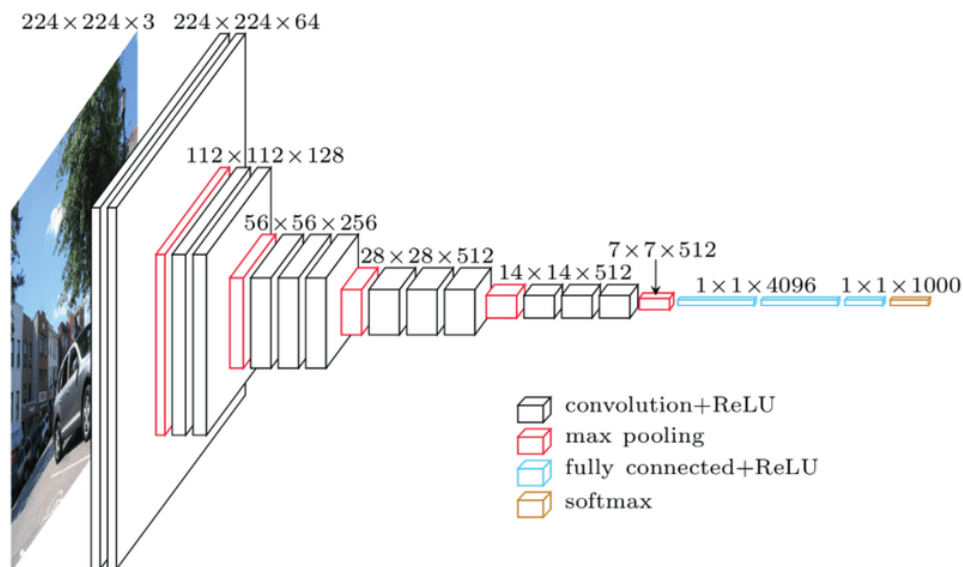
AlexNet arhitekturu konvolucijske neuronske mreže je dizajnirao Alex Krizhevsky u suradnji s Ilyom Sutskeverom i Geoffreyjem Hintonom. AlexNet se natjecao u ImageNet Large Scale Visual Recognition Challengeu 30. rujna 2012. Mreža je postigla grešku top-5 od 15,3%, što je više od 10,8% od drugoplasiranog tima. Dubina modela neuronske mreže bila je ključna za takvu dominaciju nad ostalim prijavljenim timovima. Izvedba arhitekture bila je računski skupa ali je bila izvediva zbog korištenja grafičkih procesorskih jedinica (GPU-a) tijekom treninga. Korištenjem ReLU aktivacijske funkcije je učinilo je AlexNet arhitekturu znatno bržom od ostalih.



Slika 2.3. AlexNet arhitektura<sup>3</sup>

### 2.2.3. VGGnet

VGG model, ili VGGNet, koji podržava 16 slojeva također se naziva i VGG16. VGGNet je arhitektura konvolucijske neuronske mreže koju su dizajnirali Karen Simonyan i Andrew Zisserman sa Sveučilišta u Oxfordu 2014. godine. [7] Njihov se rad uglavnom temelji na učinku dubine konvolucijske neuronske mreže na njezinu točnost. Arhitektura VGGnet-a temelj je revolucionarnih modela za prepoznavanje objekata na slikama. Razvijen kao duboka neuronska mreža koja nadmašuje osnove za mnoge zadatke i skupove podataka izvan ImageNeta. Štoviše, to je još uvijek jedna od najpopularnijih arhitektura za prepoznavanje slika.



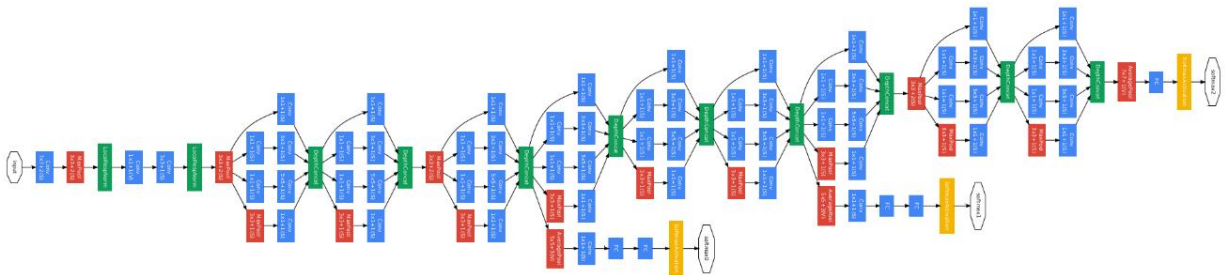
Slika 2.4. VGGnet arhitektura<sup>4</sup>

<sup>3</sup> <https://medium.com/coinmonks/paper-review-of-alexnet-caffenet-winner-in-ilsvrc-2012-image-classification-b93598314160>

<sup>4</sup> [https://www.researchgate.net/figure/VGGNet-architecture-19\\_fig2\\_333242381](https://www.researchgate.net/figure/VGGNet-architecture-19_fig2_333242381)

## 2.2.4. GoogLeNet

GoogLeNet je 22-slojna, duboka konvolucijska neuronska mreža koja je varijanta Inception Networka, duboke konvolucijske neuronske mreže koju su razvili istraživači u Googleu. Arhitektura GoogLeNet predstavljena u ImageNet Large-Scale Visual Recognition Challenge 2014 (ILSVRC14) rješavala je zadatke računalnog vida kao što su klasifikacija slike i detekcija. Danas se GoogLeNet koristi za druge zadatke računalnog vida kao što su detekcija i prepoznavanje lica, itd. Na svom početku, arhitektura GoogLeNet bila je dizajnirana kako bi bila moćna jedinica s povećanom računskom učinkovitošću u usporedbi s nekim od svojih prethodnika ili sličnih mreža stvorenih u to vrijeme. Jedna od metoda kojom GoogLeNet postiže učinkovitost je smanjenje ulazne slike, uz istovremeno zadržavanje važnih prostornih informacija.

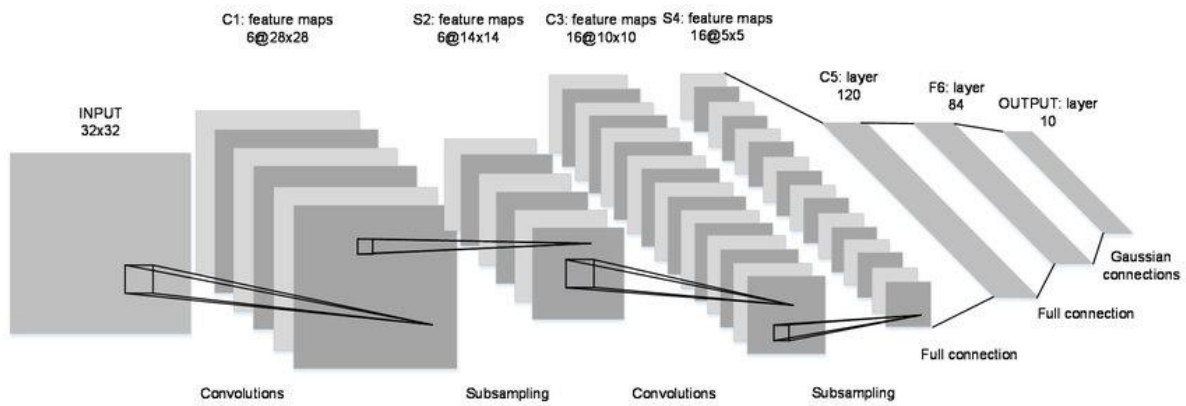


Slika 2.5. GoogLeNet arhitektura<sup>5</sup>

## 2.2.5. LeNet-5

LeNet-5 je konvolucijska struktura neuronske mreže koju su predložili Yann LeCun i suradnici 1989. godine. Bila je jedna od prvih konvolucijskih neuronskih mreža i promicala je razvoj dubokog učenja od 1988. godine. LeNet-5 je jednostavna konvolucijska neuronska mreža čije je istraživanje postiglo veliki uspjeh i pobudilo interes znanstvenika za proučavanje neuronskih mreža. Iako arhitektura neuronskih mreža s najboljim performansama danas nije ista kao LeNet-5, ova mreža je bila polazišna točka za veliki broj arhitektura neuronskih mreža, a također je donijela inspiraciju za daljnji razvoj.

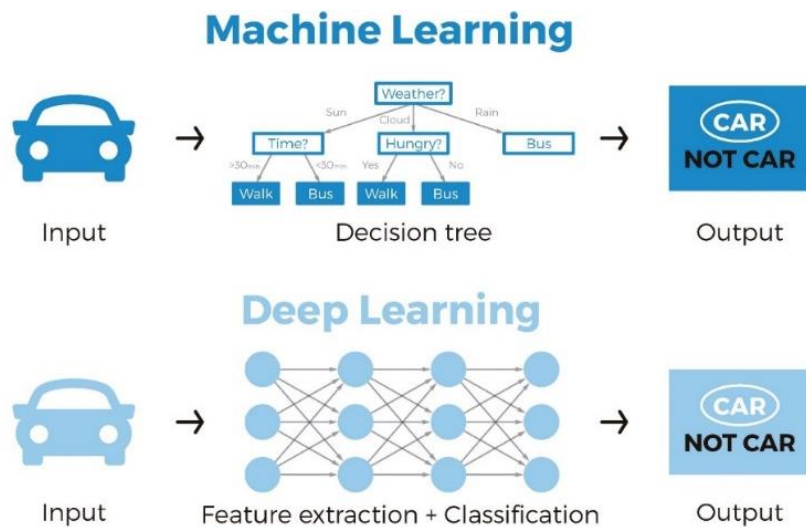
<sup>5</sup> <https://paperswithcode.com/method/googlenet>



Slika 2.6. LeNet-5 arhitektura<sup>6</sup>

### 2.3. Duboko učenje

Područje znanosti koje proučava duboke neuronske mreže naziva se duboko učenje i dio je strojnog učenja. Strojno učenje je grana šireg područja poznatog kao umjetna inteligencija. Razlika između strojnog i dubokog učenja je ta da strojno učenje istražuje algoritme koji mogu otkriti pravilnosti u podacima dok duboko učenje otkriva obrasce iz podataka učenjem značajki.



Slika 2.7. Grafički prikaz razlika između strojnog učenja i dubokog učenja<sup>7</sup>

Duboko učenje temelji se na predstavljanju podataka u složenom obliku sa visokim stupnjem apstrakcije koristeći naučene nelinearne transformacije. Metode dubokog učenja koriste se u raznim područjima: automobilskoj industriji, medicinskoj dijagnostici, računalni vid, obrada prirodnog jezika, govor i razumijevanje govora. Od 2010. godine izvanredna dostignuća dubokog

<sup>6</sup> [https://www.researchgate.net/figure/The-LeNet-5-Architecture-a-convolutional-neural-network\\_fig4\\_321586653](https://www.researchgate.net/figure/The-LeNet-5-Architecture-a-convolutional-neural-network_fig4_321586653)

<sup>7</sup> <https://www.springboard.com/blog/data-science/artificial-intelligence-vs-machine-learning-vs-deep-learning/>

učenja u rješavanju problema percepcije dovela su do revolucije u cijelom znanstvenom području. Klasifikacija prikaza, prepoznavanje govora, rukopis ili vožnja koristeći računalo sve je bliža ljudskim mogućnostima. Duboko učenje automatizira ključne korake u inženjeringu značajki strojnog učenja suzbijanjem intervencija inženjera, odnosno omogućavanjem simultanog federalnog učenja prezentacijskog sloja, radni proces strojnog učenja značajno je pojednostavljen i poboljšan, a sve zahvaljujući napretku hardverskih performansi, povećanju broja podataka, napretku softwera i algoritamskih struktura.

Intenzivno ulaganje novca i ostalih resursa od strane industrije videoigara u razvoj grafičkih kartica postupno je omogućilo suzbijanje paralelne upotrebe više procesora. Velike tvrtke počele su trenirati duboke modele na više grafičkih kartica usmjerenih na duboko učenje, a moderna industrija ulaže u čipove za treniranje dubokih neuronskih mreža koji su deset puta brži i učinkovitiji od vrhunskih grafičkih kartica. Uz velika poboljšanja hardvera za pohranu, ključni čimbenik bit će i razvoj interneta. To će omogućiti prikupljanje i distribuciju masivnih skupova podataka potrebnih za treniranje mreža. Danas velike tvrtke imaju pristup bazama podataka, koje sadrže slike, videozapise i druge podatke, koje su napravljene od strane zajednice. Posebne zasluge za razvoj dubokog učenja pripadaju bazi podataka ImageNet koja sadrži približno 1,4 milijuna slika podijeljenih u 1000 kategorija.

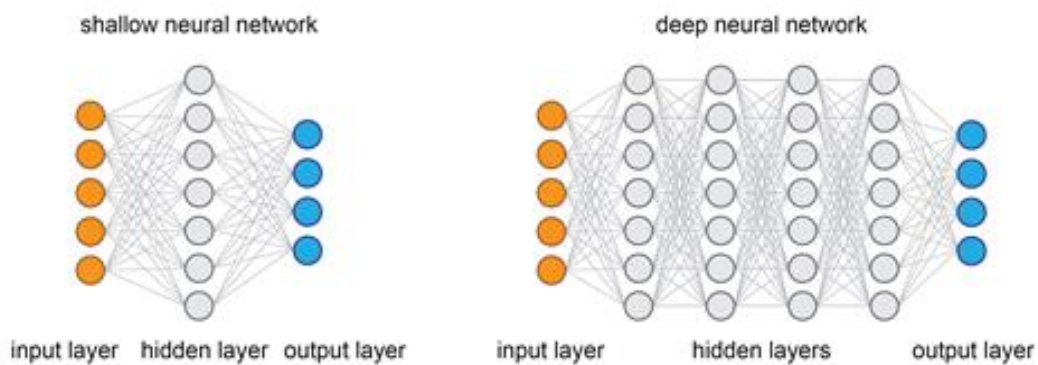
Do prije 20 godina, napredak u dubokom učenju bio je usporen nedostatkom pouzdanih metoda za učenje dubokih neuronskih mreža. Ključni problem je propagacija gradijenata kroz mnoge slojeve. To jest, povratni signal, koji je ključan za učenje neuronske mreže, gubi se kako se broj slojeva povećava. Prije otprilike deset godina napravljeno je nekoliko algoritamskih poboljšanja kako bi se poboljšalo širenje gradijenta: bolje aktivacijske funkcije neuronskih slojeva, sheme optimizacije i sheme inicijalizacije težina. To omogućuje modele dubokog učenja s deset ili više slojeva. U novije vrijeme, "Batch Normalization", "Residual Connections" i "Depthwise Repearable Convolutions" omogućuju učenje mreža sa više od tisuća slojeva.

### **2.3.1. Arhitektura dubokih neuronskih mreža**

Duboka neuronska mreža predstavlja skup neurona organiziranih u slojeve. Možemo ih promatrati i kao cikličke grafove, budući da izlaz jednog neurona predstavlja ulaz drugog neurona. Iz tog razloga ih nazivamo i unaprijednim mrežama, gdje informacije teku od ulaznih podataka, kroz sve slojeve mreže, pa sve do izlaznog sloja. Neuronska mreža u kojoj informacije teku u oba smjera naziva se rekurzivna neuronska mreža. Prvi sloj mreže naziva se i ulazni sloj, a njegova je uloga uvođenje podataka  $x^{(i)}$  u mrežu. Par  $(x^{(i)}, y^{(i)})$  u skupu podataka za učenje eksplicitno određuje klasu rezultirajućeg izlaznog vektora povezanog s ulaznim objektom  $x^{(i)}$ . Svi ostali slojevi osim izlaznog nazivaju se skrivenim slojevima jer njihove izlazne vrijednosti još uvijek ne

daju rezultate koji su zadovoljavajući. Svaki skriveni sloj može biti predstavljen neuronskim vektorom čiji je zadatak transformirati ulazne podatke dobivene prethodnim slojem i poslati rezultat neuronima sljedećeg sloja. [8]

Dubina modela neuronske mreže određena je brojem skrivenih slojeva mreže. Plitke neuronske mreže imaju samo jedan skriveni sloj dok duboke mreže imaju dva ili više skrivenih slojeva. Većinu vremena koriste se potpuno povezani slojevi, gdje su neuroni između dva susjedna sloja međusobno povezani. Slojevi su predstavljeni indeksima, počevši od nule. Izlazni sloj mreže može sadržavati samo jedan izlazni neuron ili može predstavljati vektor s višestrukim rezultatima klasifikacije.



**Slika 2.8.** Grafički prikaz razlike između plitke i duboke neuronske mreže<sup>8</sup>

<sup>8</sup> <https://medium.com/@cheahwen1997/comparison-of-insurance-cost-prediction-with-and-without-hidden-layer-in-neural-networks-1aabcc440684>

### 3. KLASIFICIRANJE GRAFIČKIH PRIKAZA KORIŠTENJEM PLITKIH NEURONSKIH MREŽA

Za usporedbu male konvolucijske neuronske mreže s nešto većom konvolucijskom mrežom koristiti će se podatkovni skup ICDAR 2019 dataset of chart images. [9] Ovo je skup podataka ručno označenih slika grafikona ekstrahiranih iz skupa Pub Med Central Open Access. Postoje 4 osnovne vrste grafikona: Bar, Line, Scatter i Box. Nekoliko je zadataka povezanih s ovim skupom podataka uključujući klasifikaciju grafikona, otkrivanje i prepoznavanje teksta, klasifikacija uloga teksta, analiza osi i analiza legende. Ukupno 4242 slike označene su za sve zadatke.

Izdanje skupa podataka sastoji se od dva indeksa u CSV formatu koji uključuju popis publikacija koje se moraju preuzeti s Pub Med Central i popis slika koje su korištene za svaki zadatak iz svake od ovih publikacija. Uz ove indekse, uključene su i python skripte koje se mogu koristiti za preuzimanje i dekomprimiranje potrebnih slika. Zatim, uključene su temeljne bilješke za svaku sliku koja je potrebna za svaki zadatak, kako u njihovom izvornom XML formatu koji koriste alati za bilješke, tako i u JSON formatu koji koriste alati za evaluaciju izvornog ICDAR CHART 2019 natjecanja.

Grafikoni su kompaktna metoda prikaza i uspoređivanja podataka. U znanstvenim publikacijama grafikoni se često koriste za sažimanje rezultata, usporedbu metodologija, naglašavanje razloga koji stoje iza ključnih aspekata znanstvenog procesa i opravdanje izbora dizajna. Automatsko izdvajanje podataka iz grafikona ključni je korak u razumijevanju namjere iza grafikona što bi moglo dovesti do boljeg razumijevanja samog dokumenta.

Zajednica je pokazala kontinuirani interes za klasificiranje vrsta grafikona, kao i za obradu grafikona za izdvajanje podataka. U proteklom desetljeću, više je aplikacija izgrađeno oko automatske obrade grafikona kao što je pronalaženje, tekstualno sažimanje grafikona, automatski redizajn grafikona, automatsku procjenu kvalitete grafikona, očuvanje grafikona iz povijesnih dokumenata, otkrivanje plagijata, bibliometrija, vizualno odgovaranje na pitanja i ubrzavanje otkrivanja novih materijala.

Prethodna natjecanja povezana s ovim područjem uključuju DeTEXT koji se koncentrirao na otkrivanje teksta u slikama i ImageCLEF natjecanje u odvajanju figura medicinskih spojeva i klasifikaciji više oznaka. Međutim, ova se natjecanja ne koncentriraju na end-to-end ekstrakciju podataka iz znanstvenih dokumenata, već ih uključuju kao podvrstu znanstvenih brojki općenito.

### 3.1. Obrada skupa podataka

Skup podataka obrađen je na način da su slike grafova organizirane u 7 kategorija: horizontal bar, horizontal box, line, pie, scatter, vertical bar i vertical box. U kategoriju horizontal bar su svrstani i stacked horizontal bar i grouped horizontal bar grafovi. U kategoriju vertical bar su svrstani i stacked vertical bar i grouped vertical bar grafovi. U kategoriju pie su svrstani i pie i donut grafovi. Broj slika po kategorijama je jednak, a podjela na trening i testing dataset je napravljena tako da se u testing datasetu nalazi 10% od količine slika koje su u training datasetu. Trainig dataset sadrži 300 slika po kategoriji, a testing dataset sadrži 30 slika po kategoriji što je ukupno 2100 slika za trening i 210 slika za testiranje.

### 3.2. Treniranje mreža

Treniranje neuronske mreže je računalno zahtjevan i dugotrajan posao. Za potrebe ovog završnog rada korištena je Google Colab platforma<sup>9</sup> i PyTorch alat. Prostor koji koristi Google Colab platforma je dijeljen sa svim ostalim Google uslugama kao što su pošta, fotografije i disk. Kako bi proces treniranja trajao kraće, slike iz dataseta uploadane su na Google Disk te su povezane sa Google Colab platformom. Velika prednost Google Colab platforme je što se može direktno pokrenuti nečiji gotov kod sa Git Hub platforme ili iz Jupyter Notebook-a što je slučaj u ovom završnom radu.

### 3.3. Plitka neuronska mreža

Kao plitka neuronska mreža, korištena je konvolucijska neuronska mreža koja je napravljena u svrhu klasificiranja CIFAR10 podatkovnog skupa u kojem se nalazi 10 klasa objekata. Kako bi se koristila navedena mreža, potrebno je na Google Colab platformi otvoriti Jupyter Notebook datoteku<sup>10</sup> koja sadrži izvorni kod. U izvornom kodu prvo se nalazi blok naredbi pomoću kojih se dodaje Pytorch alat i dodatci, zatim slijedi blok naredbi pomoću kojih se provjerava dali je na google serveru dostupna cuda jezgra pomoću koje se trenira neuronska mreža. Kako bi učitavanje podatkovnog skupa bilo brže, slike iz podatkovnog skupa su uploadane na google disk te je pomoću bloka naredbi Google Colab povezan sa Google Disk-om. U idućem bloku naredbi definirane su putanje do trening skupa i testnog skupa podataka te su određene klase podataka koje trebaju biti klasificirane. Slike u CIFAR10 podatkovnom skupu su veličine 32x32x32, odnosno 3-kanalne slike u boji, veličine 32x32 piksela. U ICDAR chart 2019 podatkovnom skupu slike su veličine 1280x960x3 odnosno 3-kanalne slike u boji, veličine 1280x960 piksela. Budući da su slike prevelike da bi ih se koristilo kao ulazne podatke u mrežu,

---

<sup>9</sup> <https://colab.research.google.com/>

<sup>10</sup> [https://pytorch.org/tutorials/beginner/blitz/cifar10\\_tutorial.html](https://pytorch.org/tutorials/beginner/blitz/cifar10_tutorial.html)



trebalo ih je smanjiti da budu prikladne veličine. U narednom bloku naredbi napravljena je transformacija te je ulazna rezolucija slika nakon transformacije 64x64 piksela. Također je određen `batch_size` koji iznosi 4 te su učitani trening i test podatkovni skupovi. Kako bi se verificiralo da je podatkovni skup učitao kako treba, u narednom bloku naredbi, prikazano je četiri slike iz podatkovnog skupa zajedno s klasama kojima pripadaju. Idući blok naredbi donosi definiciju konvolucijske neuronske mreže. Konvolucijska neuronska mreža koja je korištena u zadatku završnog rada sastoji se od dva konvolucijska sloja, jednog pooling sloja te tri potpuno povezana sloja. Sve ukupno mreža sadrži 338231 parametara. Prvi konvolucijski sloj, koji se primjenjuje direktno na ulaznu sliku veličine 64x64 piksela, ima kao ulaz 3 kanala dok na izlazu ima 6 kanala. Kernel korišten u prvom sloju je veličine 5x5 piksela sa strideom 1. Rezultat Prvog konvolucijskog sloja su 6 kanala veličine 60x60 piksela. Nakon prvog sloja neuronske mreže primijenjena je ReLU aktivacijska funkcija. Nakon toga slijedi max pooling sloj koji ima kernel veličine 2x2 piksela te stride 2. Izlaz iz max pooling sloja je veličine 30x30 piksela sa 6 kanala. Nakon toga slijedi drugi konvolucijski sloj koji kao ulaz ima 6 kanala dok na izlazu ima 16 kanala. U ovom konvolucijskom sloju također je korišten kernel veličine 5x5 piksela te je također na njega primijenjena ReLU aktivacijska funkcija. Kao rezultat drugog konvolucijskog sloja dobiva se 16 kanala veličine 13x13 piksela. Nakon drugog konvolucijskog sloja slijedi flatten funkcija koja poravnava unos preoblikujući ga u jednodimenzionalni tenzor odnosno vektor. Redosljed elemenata u unosu je nepromijenjen. Nakon poravnavanja slijede tri potpuno povezana sloja gdje prvi na ulazu ima 16x13x13 neurona dok na izlazu ima 120 neurona te je primijenjena ReLU aktivacijska funkcija. Drugi potpuno povezani sloj na ulazu ima 120 neurona, a na izlazu 84 neurona također sa ReLU aktivacijskom funkcijom. Na kraju treći potpuno povezani sloj ima na ulazu 84 neurona i kao izlaz ima 7 parametara odnosno klasificira ulazni skup slika u 7 kategorija. Blokovski prikaz plitke konvolucijske neuronske mreže prikazan je na slici 3.1., a izvorni kod definicije plitke konvolucijske neuronske mreže izgleda ovako:

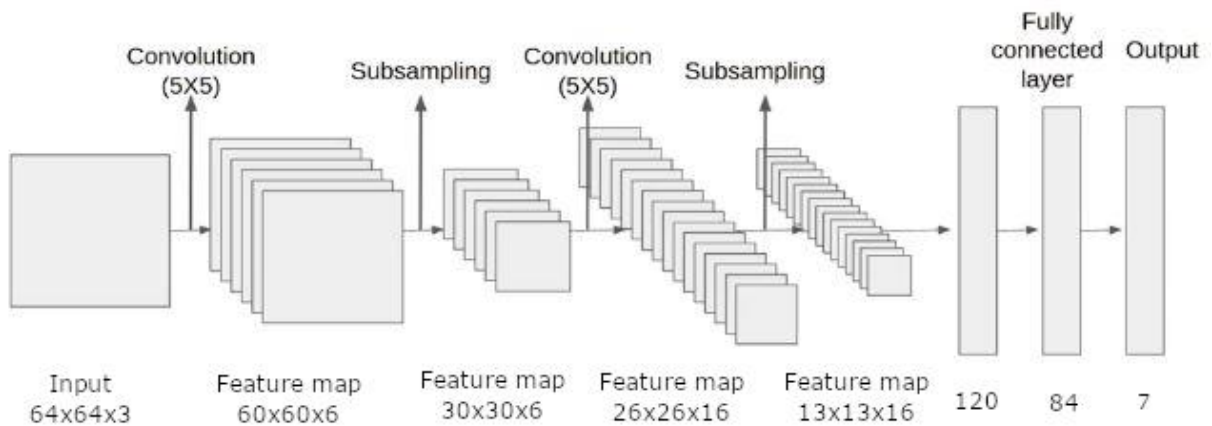
```
class Net(nn.Module):
    def __init__(self):
        super().__init__()
        self.conv1 = nn.Conv2d(3, 6, 5)
        self.pool = nn.MaxPool2d(2, 2)
        self.conv2 = nn.Conv2d(6, 16, 5)
        self.fc1 = nn.Linear(16 * 13 * 13, 120)
        self.fc2 = nn.Linear(120, 84)
        self.fc3 = nn.Linear(84, 7)

    def forward(self, x):
        x = self.pool(F.relu(self.conv1(x)))
```

```

x = self.pool(F.relu(self.conv2(x)))
x = torch.flatten(x, 1)
x = F.relu(self.fc1(x))
x = F.relu(self.fc2(x))
x = self.fc3(x)
return x

```



**Slika 3.1.** Blokovski prikaz plitke konvolucijske neuronske mreže<sup>11</sup>

Nakon definiranja konvolucijske neuronske mreže, za provjeru, isprintan je broj parametara mreže te struktura konvolucijske neuronske mreže. Nakon što je utvrđeno da je sve u redu, određuje se kriterij optimizacije i optimizacijska funkcija. U idućem bloku naredbi priređuju se varijable za mjerenje vremena koje je potrebno za treniranje konvolucijske neuronske mreže, određuje se broj epoha te se pokreće treniranje neuronske mreže. Nakon što se mreža istrenirala, sprema se te se ponovno pokreće kako bi se provjerila točnost klasificiranja slika iz podatkovnog skupa po klasama. Kako bi se svi blokovi naredbi pokrenuli potrebno je povezati se na Google server pritiskom na „Connect“ u gornjem desnom uglu zatim je potrebno na alatnoj traci kliknuti na karticu „Runtime“ te kliknuti na „run all“.

### 3.4. Duboka neuronska mreža

Za usporedbu sa plitkom neuronskom mrežom korištena je VGG-11 konvolucijska neuronska mreža jer su dublje mreže preteške za besplatnu verziju Goole Colab platforme. VGG-11 je unaprijed istrenirani model, na skupu podataka i sadrži težine koje predstavljaju značajke bilo kojeg skupa podataka na kojem je istreniran. Korištenje unaprijed istreniranog modela štedi vrijeme. Već je dosta vremena i računalnih resursa potrošeno da se nauči mnogo značajki i model će vjerojatno imati koristi od toga. Kako bi se implementirala navedena konvolucijska neuronska

<sup>11</sup> <https://www.analyticsvidhya.com/blog/2021/03/the-architecture-of-lenet-5/>

mreža korišten je već postojeća Jupyter Notebook datoteka<sup>12</sup> koja sadrži izvorni kod implementacije VGG-11 neuronske mreže. Blokovi naredbi su poslagani istim redoslijedom kao i kod male mreže samo što je konvolucijska neuronska mreža drugačija. Broj 11 iza ključne riječi VGG označava broj slojeva u modelu. VGG modeli uzimaju kao ulaz sliku veličine 224 x 224 piksela koja je u 3 kanala (RGB). Kako bi skup podataka koji se koristi za zadatak završnog rada bio pogodan za ovu neuronsku mrežu napravljena je transformacija ulaznih podataka tako da se slike veličine 3x1280x960 odnosno 3-kanalne slike u boji, veličine 1280x960 piksela smanje na rezoluciju od 224x224 piksela. VGG-11 mreža sastoji se od 128800519 parametara što je preko 380 puta više nego količina parametara plitke neuronske mreže korištene u ovome zadatku. Osim veličine slika u VGG-11 konvolucijskoj neuronskoj mreži bilo je potrebno modificirati zadnji sloj kako bi se ulazni skup slika mogao klasificirati u 7 kategorija.

U konvolucijskim slojevima VGG-11 mreže filter je veličine 3x3 piksela, elementi matrice uzimaju se jedan po jedan za izvođenje odgovarajuće operacije te je korištena ReLU aktivacijska funkcija. Max pooling sloj vraća piksel s maksimalnom vrijednošću iz skupa piksela unutar kernela. Korak konvolucije je fiksiran na jedan piksel. Izabrana je ReLU aktivacijska funkcija jer je jednostavnije računanje, pogodna je za velike neuronske mreže jer smanjuje vrijeme treniranja i testiranja te također smanjuje potrošnju računalnih resursa. To omogućuje treniranje većih mreža s više parametara sa istim hardverom odnosno s istim resursima. U ovoj mreži postoje tri potpuno povezana sloja od koji su dva sloja sa po 4096 neurona dok se na izlazu trećega potpuno povezanoga sloja nalazi samo sedam neurona koji odgovaraju broju klasa u koje se ulazne slike trebaju klasificirati

VGG-11 konvolucijska neuronska mreža sastoji se od ukupno 11 težinskih slojeva od kojih je 8 konvolucijskih slojeva te 3 potpuno povezana sloja. Svi konvolucijski slojevi imaju isti kernel koji je veličine 3x3 piksela sa strideom 1, također i svi max pooling slojevi imaju isti kernel koji je veličine 2x2 piksela sa strideom 2. Prvi konvolucijski sloj, koji se primjenjuje direktno na ulaznu sliku veličine 224x224 piksela, ima kao ulaz 3 kanala. Rezultat prvog konvolucijskog sloja su 64 kanala veličine 224x224 piksela. Nakon toga slijedi max pooling sloj. Izlaz iz max pooling sloja je veličine 112x112 piksela sa 64 kanala. Drugi konvolucijski sloj prima izlazne podatke od prvoga max pooling sloja i kao rezultat daje slike veličine 112x112 piksela sa 128 kanala. Nakon drugog konvolucijskog sloja slijedi drugi max pooling sloj koji za izlaz daje slike veličine 56x56 piksela sa 128 kanala. Treći konvolucijski sloj prima podatke od drugog max pooling sloja te na izlazu daje 256 kanalne slike veličine 56x56 piksela. Četvrti konvolucijski sloj prima podatke od trećeg

---

<sup>12</sup> [https://colab.research.google.com/github/bentrevett/pytorch-image-classification/blob/master/4\\_vgg.ipynb](https://colab.research.google.com/github/bentrevett/pytorch-image-classification/blob/master/4_vgg.ipynb)

konvolucijskog sloja i na izlazu daje također 256 kanalne slike veličine 56x56 piksela. Nakon četvrtog konvolucijskog sloja slijedi treći max pooling sloj koji prima slike od četvrtog konvolucijskog sloja te na izlazu daje slike veličine 28x28 piksela sa 256 kanala. Peti konvolucijski sloj prima slike od trećeg max pooling sloja i na izlazu daje slike veličine 28x28 piksela sa 512 kanala. Šesti konvolucijski sloj prima slike od petog konvolucijskog sloja i daje isti izlaz. Nakon šestog konvolucijskog sloja slijedi četvrti max pooling sloj koji prima podatke od šestog konvolucijskog sloja i na izlazu daje slike veličine 14x14 piksela sa 512 kanala. Sedmi i osmi konvolucijski sloj su isti, oba daju na izlazu slike veličine 14x14 piksela sa 512 kanala. Sedmi konvolucijski sloj prima slike od četvrtog max pooling sloja dok osmi konvolucijski sloj prima slike od sedmog konvolucijskog sloja. Nakon osmog konvolucijskog sloja slijedi peti max pooling sloj koji na izlazu daje slike veličine 7x7 piksela sa 512 kanala. Na kraju se nalaze tri potpuno povezana sloja, kod prvog potpuno povezanog sloja kao ulaz dolazi izlaz od osmoga konvolucijskog sloja, 7x7x512 dok na izlazu ima 4096 parametara. Drugi potpuno povezani sloj prima podatke od prvog potpuno povezanog sloja te na izlazu daje također 4096 parametara. Treći potpuno povezani sloj je ujedno i zadnji sloj koji na ulazu ima 4096 parametara dok na izlazu ima 7 parametara koji odgovaraju klasama grafikona. Kako bi se svi blokovi naredbi pokrenuli potrebno je povezati se na Google server pritiskom na „Connect“ u gornjem desnom uglu zatim je potrebno na alatnoj traci kliknuti na karticu „Runtime“ te kliknuti na „run all“. Blokovski prikaz VGG-11 konvolucijske neuronske mreže prikazan je na slici 3.2., a izvorni kod definicije VGG-11 konvolucijske neuronske mreže izgleda ovako:

```
class VGG11(nn.Module):
    def __init__(self, in_channels, num_classes=1000):
        super(VGG11, self).__init__()
        self.in_channels = in_channels
        self.num_classes = num_classes

        # convolutional layers
        self.conv_layers = nn.Sequential(
            nn.Conv2d(self.in_channels, 64, kernel_size=3, padding=1),
            nn.ReLU(),
            nn.MaxPool2d(kernel_size=2, stride=2),
            nn.Conv2d(64, 128, kernel_size=3, padding=1),
            nn.ReLU(),
            nn.MaxPool2d(kernel_size=2, stride=2),
            nn.Conv2d(128, 256, kernel_size=3, padding=1),
            nn.ReLU(),
            nn.Conv2d(256, 256, kernel_size=3, padding=1),
            nn.ReLU(),
            nn.MaxPool2d(kernel_size=2, stride=2),
            nn.Conv2d(256, 512, kernel_size=3, padding=1),
```

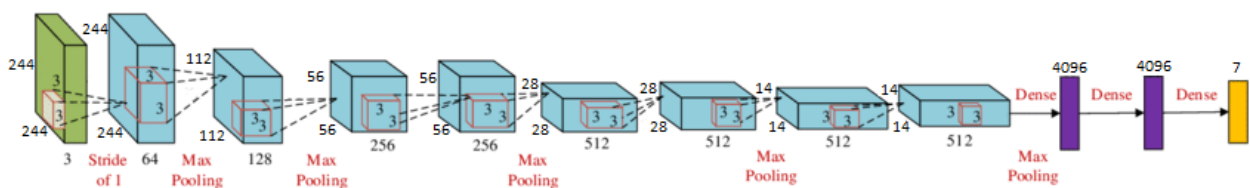
```

nn.ReLU(),
nn.Conv2d(512, 512, kernel_size=3, padding=1),
nn.ReLU(),
nn.MaxPool2d(kernel_size=2, stride=2),
nn.Conv2d(512, 512, kernel_size=3, padding=1),
nn.ReLU(),
nn.Conv2d(512, 512, kernel_size=3, padding=1),
nn.ReLU(),
nn.MaxPool2d(kernel_size=2, stride=2)
)

# fully connected linear layers
self.linear_layers = nn.Sequential(
    nn.Linear(in_features=512*7*7, out_features=4096),
    nn.ReLU(),
    nn.Dropout2d(0.5),
    nn.Linear(in_features=4096, out_features=4096),
    nn.ReLU(),
    nn.Dropout2d(0.5),
    nn.Linear(in_features=4096, out_features=self.num_classes)
)

def forward(self, x):
    x = self.conv_layers(x)
    # flatten to prepare for the fully connected layers
    x = x.view(x.size(0), -1)
    x = self.linear_layers(x)
    return x

```



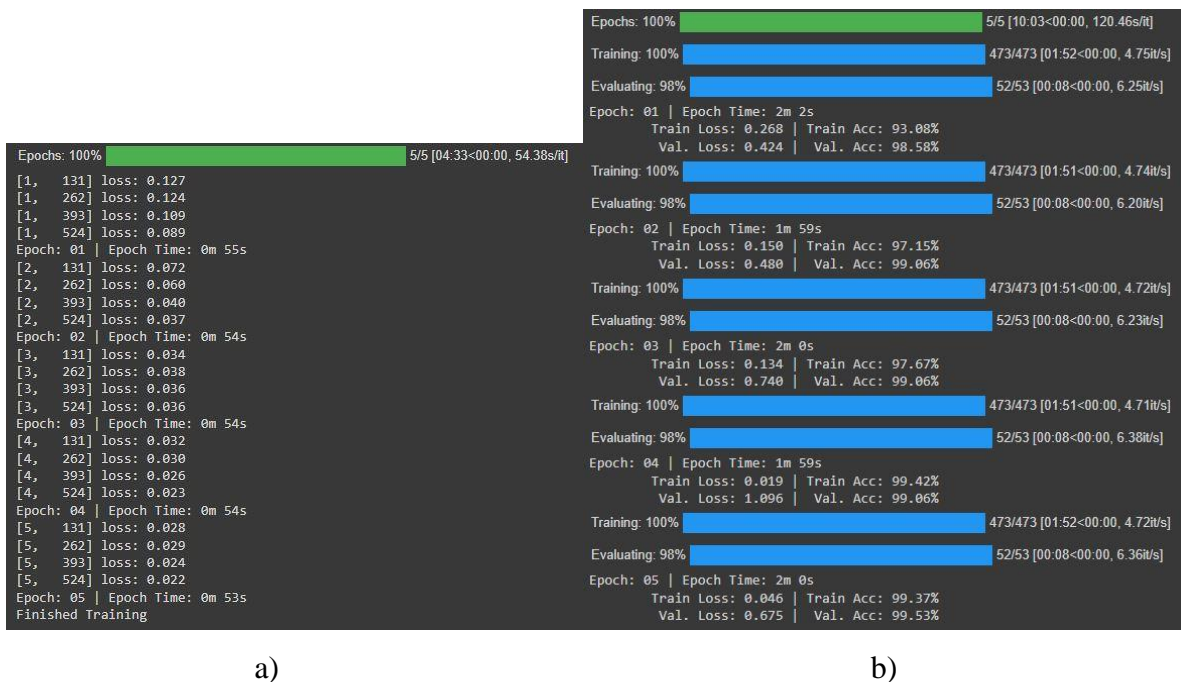
Slika 3.2. Blokovski prikaz VGG-11 konvolucijske neuronske mreže<sup>13</sup>

### 3.5. Rezultati zadatka

Kako bi se osigurao prosjek rezultata svaka neuronska mreža trenirana je 5 puta te je nakon svakog puta zabilježen rezultat odnosno postotak sigurnosti klasificiranja slika. Vrijeme treniranja velike i male neuronske mreže prikazano je na slici 3.3. Može se primjetiti kako je za treniranje

<sup>13</sup> [https://www.researchgate.net/figure/VGG-11-structure-diagram\\_fig1\\_333461196](https://www.researchgate.net/figure/VGG-11-structure-diagram_fig1_333461196)

male mreže potrebno duplo manje vremena jer sadrži manje parametara. Na velikoj mreži je dobar dio parametara već istreniran no onih koji su ne istrenirani ima više nego u maloj mreži. Valja napomenuti kako vrijeme treniranja mreže ovisi o hardveru odnosno specifikacijama računala na kojem se trenira neuronska mreža. U ovom slučaju, pošto su mreže trenirane na Google Colab platformi, nije moguće znati kakve su specifikacije računala.

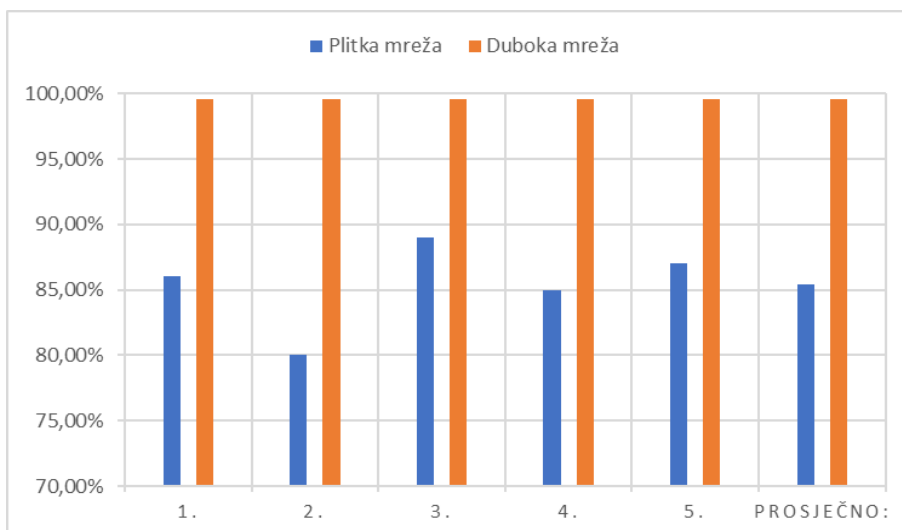


**Slika 3.3.** Prikaz potrebnog vremena za treniranje a) male mreže i b) velike mreže

Slika se može smatrati uspješno klasificiranom ako je mreža točno pogodila i ako je mreža sigurna u svoj odgovor. Postotak sigurnosti ne bi trebao biti manji od 80% za svaku sliku. Rezultati testiranja zapisani su u tablici 3.1. te su prikazani grafički na slici 3.4.

**Tablica 3.1.** Rezultati treniranja konvolucijskih neuronskih mreža.

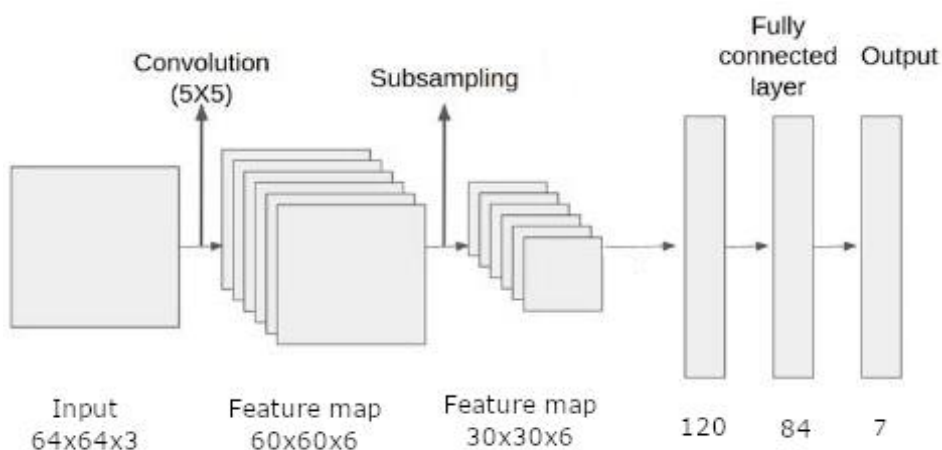
Broj iteracije	Mreža	
	Plitka mreža	Duboka mreža
1.	86%	99,53%
2.	80%	99.53%
3.	89%	99.53%
4.	85%	99.53%
5.	87%	99.53%
Prosječno:	85,4%	99.53%



**Slika 3.4.** Grafički prikaz rezultata treniranja duboke i plitke mreže

Plitka neuronska mreža klasificira slike u kategorije sa prosječnom točnošću od 85,4% što je zadovoljavajući rezultat dok s druge strane duboka mreža klasificira slike u kategorije sa prosječnom točnošću od 99.53%. Kod treniranja duboke mreže dobiveni su svaki puta isti rezultati jer je dobar dio mreže već istreniran pa se nije puno parametara moralo ponovno trenirati.

Naredni koraci u zadatku bili su da se u plitkoj neuronskoj mreži kombiniraju slojevi sve dok se ne dobije zadovoljavajuća neuronska mreža za zadani zadatak. Za početak uklonjen je drugi konvolucijski sloj iz neuronske mreže. Na slici 3.5. prikazan je blokovski prikaz plitke neuronske mreže sa uklonjenim jednim konvolucijskim slojem.



**Slika 3.5.** Blokovski prikaz konvolucijske mreže s uklonjenim prvim konvolucijskim slojem

Treniranje mreže provedeno je 5 puta kao u prvom koraku. Rezultati treniranja plitke konvolucijske neuronske mreže sa jednim konvolucijskim slojem manje je prikazani su u tablici 3.2. te su prikazani grafički na slici 3.6.

**Tablica 3.2.** Rezultati treniranja plitke konvolucijske neuronske mreže

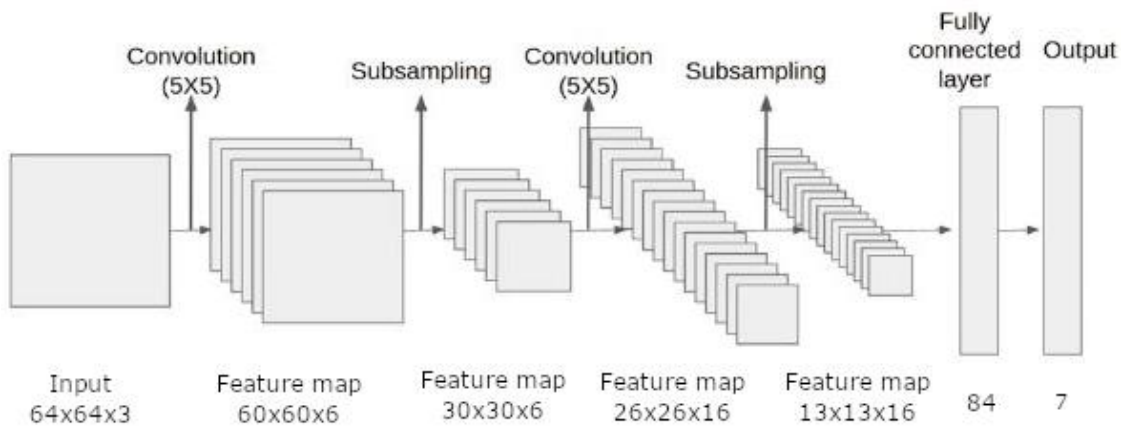
Broj iteracije	Mreža	Plitka mreža
1.		81%
2.		84%
3.		77%
4.		85%
5.		83%
Prosječno:		82%



**Slika 3.6.** Grafički prikaz rezultata treniranja plitke mreže s uklonjenim drugim konvolucijskim slojem

Oduzimanje jednoga konvolucijskoga sloja je dovelo do toga da mreža klasificira slike u kategorije sa prosječnom točnošću od 82% što je i dalje zadovoljavajuće. Nakon 5 treninga s uklonjenim drugim konvolucijskim slojem, vraćen je taj konvlucijski sloj te je ukonjen prvi potpuno povezani sloj iz plitke konvolucijske neuronske mreže. Na slici 3.7. prikazan je blokovski prikaz plitke neuronske mreže sa uklonjenim prvim potpuno povezanim slojem slojem.



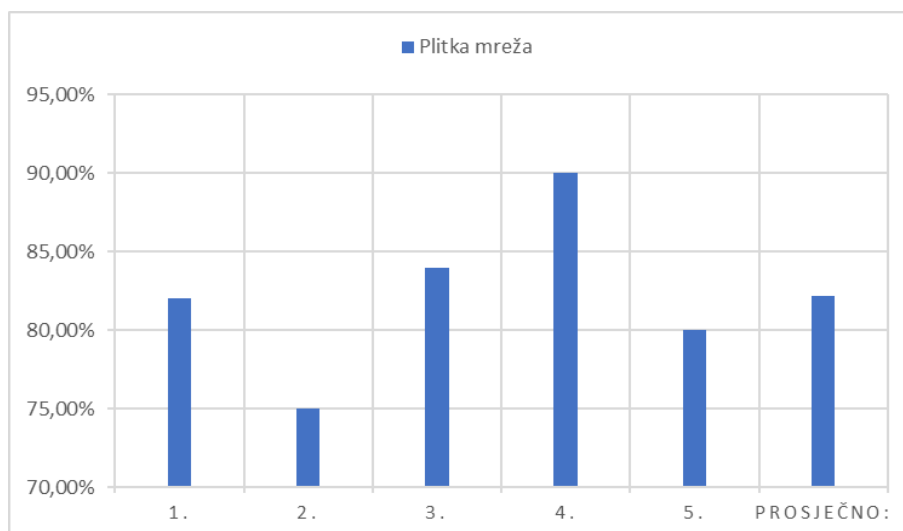


**Slika 3.7.** Blokovski prikaz plitke konvolucijske mreže s uklonjenim prvim potpuno povezanim slojem

Rezultati treniranja plitke konvolucijske neuronske mreže sa uklonjenim prvim potpuno povezanim slojem prikazani su u tablici 3.3. te su prikazani grafički na slici 3.8.

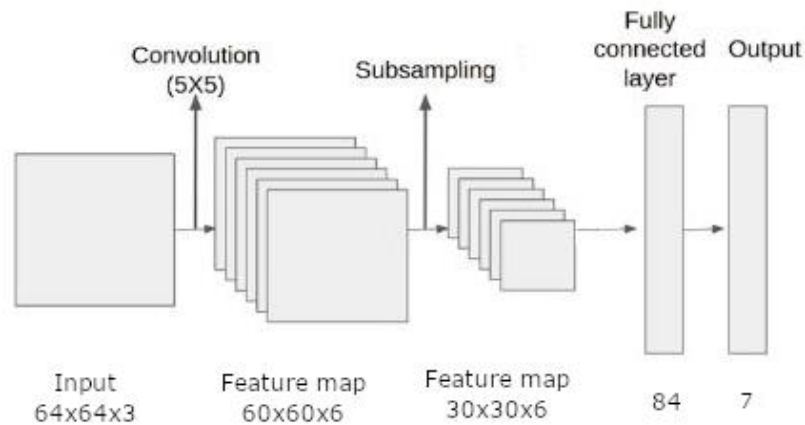
**Tablica 3.3.** Rezultati treniranja plitke konvolucijske neuronske mreže

Broj iteracije	Mreža	Plitka mreža
1.		82%
2.		75%
3.		84%
4.		90%
5.		80%
Prosječno:		82,2%



**Slika 3.8.** Grafički prikaz rezultata treniranja plitke mreže s uklonjenim prvim potpuno povezanim slojem

Plitka konvolucijska mreža sa uklonjenim prvim konvolucijskim slojem klasificira slike s prosječnom tačnošću od 82.2% što je zadovoljavajući rezultat. Nakon 5 treninga sa uklonjenim prvim pozpuno povezanim slojem, ukonjen je opet drugi konvolucijski sloj. Izgled plitke konvolucijske neuronske mreže bez drugog konvolucijskog sloja i bez prvog potpuno povezanog sloja prikazan je na slici 3.9.

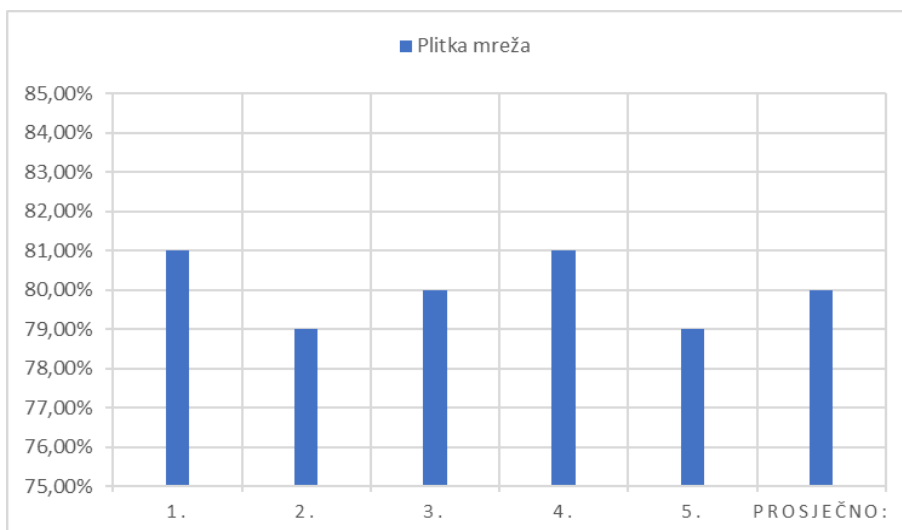


**Slika 3.9.** Blokovski prikaz konvolucijske mreže s uklonjenim drugim konvolucijskim slojem i prvim potpuno povezanim slojem

Rezultati treniranja plitke konvolucijske neuronske mreže s uklonjenim drugim konvolucijskim slojem i prvim potpuno povezanim slojem prikazani su u tablici 3.4. te su prikazani grafički na slici 3.10.

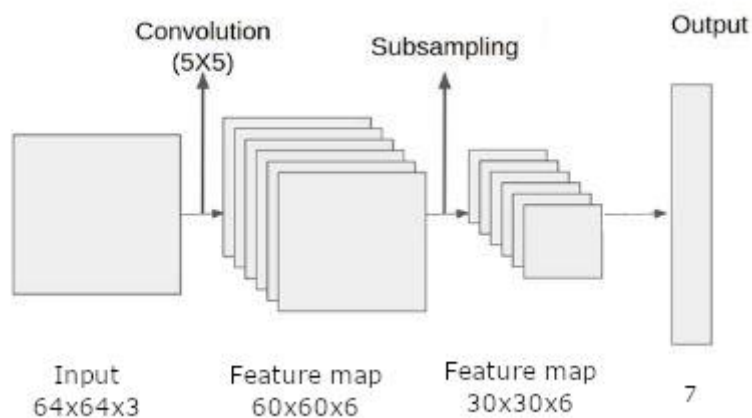
**Tablica 3.4.** Rezultati treniranja plitke konvolucijske neuronske mreže

Broj iteracije \ Mreža	Mreža	Plitka mreža
1.		81%
2.		79%
3.		80%
4.		81%
5.		79%
Prosječno:		80%



**Slika 3.10.** *Grafički prikaz rezultata treniranja plitke mreže s uklonjenim drugim konvolucijskim slojem i prvim potpuno povezanim slojem*

Oduzimanje jednog konvolucijskog sloja i jednog potpuno povezanog sloja dovelo je do toga da su rezultati klasificiranja slika u kategorije na samoj granici zadovoljavanja uvjeta pod kojima se mreža može smatrati dobra. U posljednjem koraku uklonjen je drugi potpuno povezani sloj tako da je ostao samo jedan konvolucijski sloj i jedan potpuno povezani sloj. Izgled plitke konvolucijske neuronske mreže sa uklonjenim drugim konvolucijskim slojem i sa uklonjenim prvim i drugim potpuno povezanim slojevima prikazan je na slici 3.11.

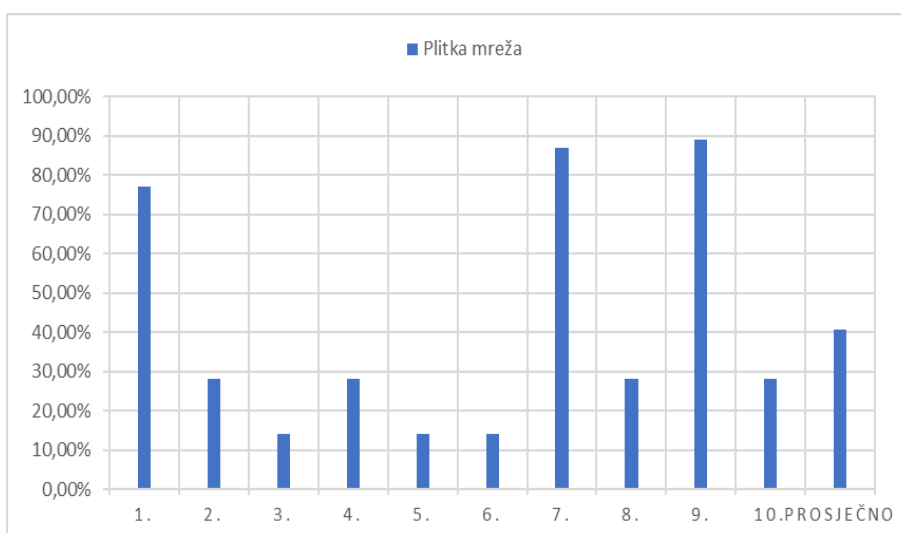


**Slika 3.11.** *Blokovski prikaz konvolucijske mreže s uklonjenim drugim konvolucijskim slojem i prvim i drugim potpuno povezanim slojem*

Sa ovakvom konvolucijskom mrežom provedeno je 10 treniranja i rezultati se nalaze u tablici 3.5. te su prikazani grafički na slici 3.12.

**Tablica 3.5.** Rezultati treniranja plitke konvolucijske neuronske mreže

Broj iteracije \ Mreža	Plitka mreža
1.	77%
2.	28%
3.	14%
4.	28%
5.	14%
6.	14%
7.	87%
8.	28%
9.	89%
10.	28%
Prosječno	40,7%

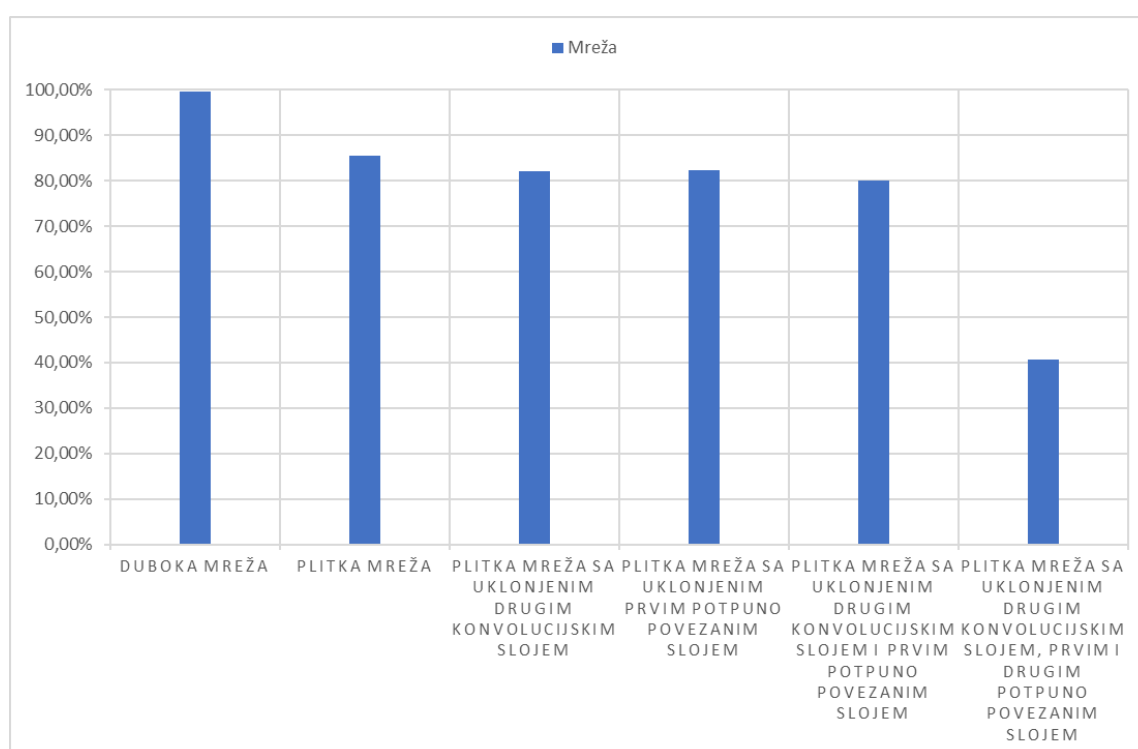


**Slika 3.12.** Grafički prikaz rezultata treniranja plitke mreže s uklonjenim drugim konvolucijskim slojem, prvim i drugim potpuno povezanim slojem

Iz rezultata je vidljivo da je od 10 treniranja mreža samo dva puta klasificirala slike u kategorije s točnošću preko 80%. Da se zaključiti kako konvolucijska neuronska mreža s jednim konvolucijskim slojem, jednim max pooling slojem i jednim potpuno povezanim slojem nije dovoljno dobra za klasificiranje slika u kategorije. Svi dobiveni rezultati sumirani su u tablici 3.6. te su prikazani grafički na slici 3.13.

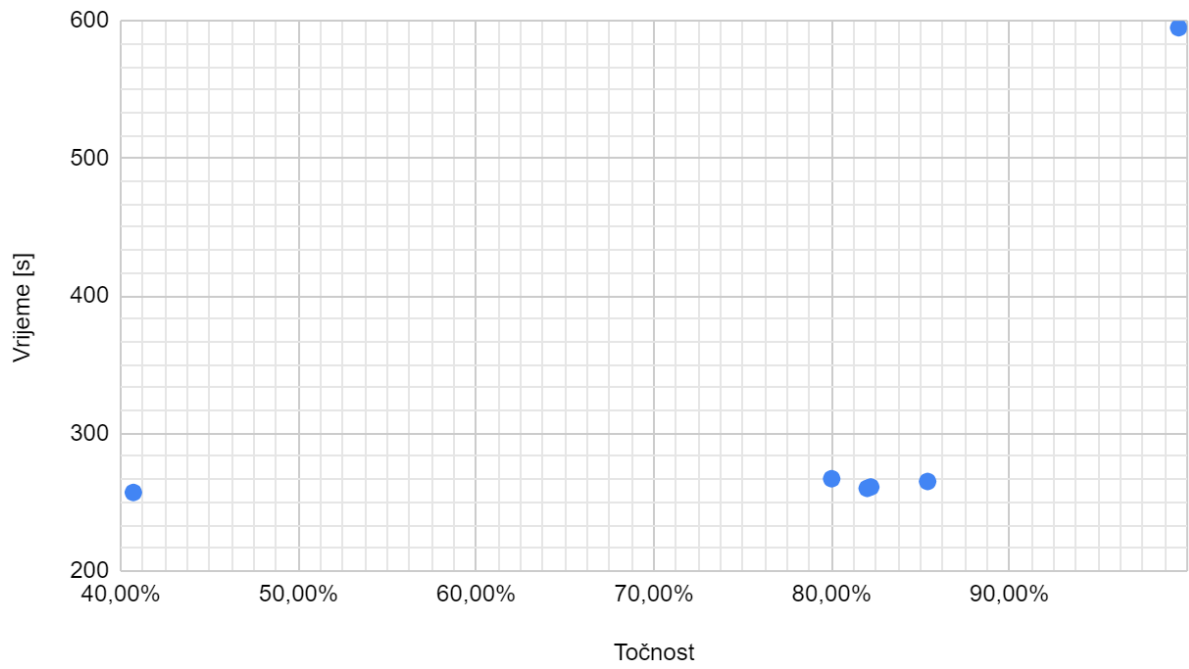
**Tablica 3.6. Rezultati treniranja svih konvolucijskih mreža**

Mreža	Prosječni postotak točnosti	Trajanje treniranja u sekundama
Duboka mreža	99,53%	595
Plitka mreža	85,4%	265
Plitka mreža sa uklonjenim drugim konvolucijskim slojem	82%	260
Plitka mreža sa uklonjenim prvim potpuno povezanim slojem	82,2%	261
Plitka mreža sa uklonjenim drugim konvolucijskim slojem i prvim potpuno povezanim slojem	80%	267
Plitka mreža sa uklonjenim drugim konvolucijskim slojem, prvim i drugim potpuno povezanim slojem	40,7%	257



**Slika 3.13. Grafički prikaz rezultata treniranja svih korištenih mreža**

Na slici 3.14. prikazan je prosječan postotak točnosti klasificiranja slika u odnosu na vrijeme treniranja svih mreža prikazano u sekundama. Može se primjetiti kako je za treniranje duboke mreže potrebno skoro tri puta više vremena nego za malu mrežu i sve njene kombinacije, također i točnost klasificiranja duboke mreže je znatno veća. Treniranje male mreže je približno jednako za svaku verziju koja je korištena u ovom radu. Točnost klasificiranja male mreže približno jednaka za sve njene kombinacije osim za najmanju verziju male mreže.



**Slika 3.14.** *Prikaz vremena potrebnog za treniranje mreža u odnosu na prosječan postotak točnosti klasificiranja*

## 4. ZAKLJUČAK

Plitke i duboke konvolucijske neuronske mreže sposobne su aproksimirati bilo koju funkciju. Za istu razinu točnosti, dublje mreže mogu biti puno učinkovitije u smislu izračuna i broja parametara. Dublje mreže su u stanju stvoriti duboke reprezentacije, na svakom sloju mreža uči novi, apstraktniji prikaz ulaza. Plitke mreže jednostavno nemaju toliku moć objašnjavanja kao duboke mreže. Izazov s dubokim mrežama je u tome što nema dovoljno velikog skupa podataka za treniranje da bi mreža radila kako treba. Ako imamo mrežu s jednim slojem, broj neurona koji je potreban da bi se poklopio s višeslojnom mrežom kombinatorno je velik. Uz više slojeva, svaki sloj može stvoriti sve više apstraktnih značajki i koncepta. Kao u ljudskom mozgu, izlaz iz očiju ulazi u mozak i prolazi kroz detektore značajki vrlo niske razine. Kako se signal kreće naprijed kroz slojeve, značajke postaju apstraktnije, od jednostavnih detektora linija do raznih vrsta pokretnih objekata. Tako je i s neuronskim mrežama. Dubina daje mogućnost apstrahiranja relativno apstraktnih konceptata u gornjim slojevima što uvelike poboljšava sposobnost mreže za klasifikaciju sve dok ima dovoljno podataka.

## LITERATURA

- [1] umjetna inteligencija. Hrvatska enciklopedija, mrežno izdanje. Leksikografski zavod Miroslav Krleža, 2021. Pristupljeno 21. 3. 2022. <http://www.enciklopedija.hr/Natuknica.aspx?ID=63150>
- [2] I.Goodfellow, Y.Bengio, A.Courville, Deep Learning, MIT Press, 2016
- [3] Strojno učenje, Osvježimo znanje, N. Bolf, Sveučilište u Zagrebu Fakultet kemijskog inženjerstva i tehnologije, Pristupljeno 21.3.2022.  
<https://hrcak.srce.hr/file/382926>
- [4] Novaković, Branko ; Majetić, Dubravko ; Široki, Mladen (1998) Umjetne neuronske mreže. = Artificial neural networks. Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje, Zagreb, -. ISBN 953-6313-17-0
- [5] Umjetna neuronska mreža, Pristupljeno 23.3.2022.  
[https://hr.wikipedia.org/wiki/Umjetna\\_neuronska\\_mre%C5%Bea](https://hr.wikipedia.org/wiki/Umjetna_neuronska_mre%C5%Bea)
- [6] Convolutional Neural Networks, Pristupljeno 23.3.2022.  
<https://cs231n.github.io/convolutional-networks/>
- [7] VGG16 – Convolutional Network for Classification and Detection, Pristupljeno 23.3.2022.  
<https://neurohive.io/en/popular-networks/vgg16/>
- [8] Alex Krizhevsky, Ilya Sutskever. Geoffrey E. Hinton - ImageNet Classification with Deep Convolutional Neural Networks, Pristupljeno 23.3.2022.  
<http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf>
- [9] Competition on Harvesting Raw Tables from Infographics, Pristupljeno 23.3.2022.  
[https://chartinfo.github.io/index\\_2019.html](https://chartinfo.github.io/index_2019.html)



## SAŽETAK

Duboke neuronske mreže su standardni alat koji pomaže u rješavanju problema kao što su detekcija objekata, raspoznavanje oblika i klasificiranje podataka. Potrebna je određena količina znanja kako bi se duboka neuronska mreže istrenirala za željenu radnju. Precizno označeni podatkovni skupovi uvelike pomažu pri učenju neuronske mreže. Što više podataka za učenje prosljedimo neuronskoj mreži to će ona bolje raditi svoj zadatak za koji je istrenirana.

U ovom radu opisane su osnove umjetne inteligencije i strojnog učenja. Istražene su vrste neuronskih mreža i razlika između dubokih i plitkih neuronskih mreža. U svrhu istraživanja razlike učinkovitosti između plitkih i dubokih konvolucijskih neuronskih mreža odabrana je mala konvolucijska neuronska mreža koja se sastoji od dva konvolucijska sloja te tri potpuno povezana sloja i VGG-11 konvolucijska neuronska mreža koja se sastoji od osam konvolucijskih i tri potpuno povezana sloja.

### **Ključne riječi:**

duboke konvolucijske neuronske mreže, konvolucijske neuronske mreže,  
plitke konvolucijske neuronske mreže, strojno učenje, umjetna inteligencija, umjetne neuronske mreže

## **ABSTRACT**

### **CLASSIFICATION OF GRAPHICS USING SHALLOW NEURAL NETWORKS**

Deep neural networks are a standard tool to help solve problems such as object detection, shape recognition, and data classification. A certain amount of knowledge is required to train a deep neural network for a desired action. Precisely labeled datasets greatly aid neural network learning. The more training data we feed the neural network, the better it will perform at the task for which it was researched.

This paper describes the basics of artificial intelligence and machine learning. The types of neural networks and the difference between deep and shallow neural networks are explored. In order to investigate the difference in efficiency between shallow and deep convolutional neural networks, a small convolutional neural network consisting of two convolutional layers and three fully connected layers and a VGG-11 convolutional neural network consisting of eight convolutional and three fully connected layers were selected.

#### **Keywords:**

deep convolutional neural networks, convolutional neural networks, shallow convolutional neural networks, machine learning, artificial intelligence, artificial neural networks

## **ŽIVOTOPIS**

Filip Hofer rođen je 17. kolovoza 2000. godine u Našicama. 2019. godine po završetku matematičko-prirodoslovne III. Gimnazije u Osijeku upisuje preddiplomski sveučilišni studij na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku, smjer elektrotehnika.

---

Potpis

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**IZJAVA O ORIGINALNOSTI RADA**

Osijek, 21.09.2022.

**Ime i prezime studenta:**

Filip Hofer

**Studij:**

Preddiplomski sveučilišni studij Elektrotehnika i informacijska tehnologija

**Mat. br. studenta, godina upisa:**

4680, 22.07.2019.

**Turnitin podudaranje [%]:**

13

Ovom izjavom izjavljujem da je rad pod nazivom: **Klasificiranje grafičkih prikaza korištenjem plitkih neuronskih mreža**

izrađen pod vodstvom mentora izv. prof. dr.sc. Josip Job

i sumentora ,

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija. Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis studenta:

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK****Obrazac Z1P - Obrazac za ocjenu završnog rada na preddiplomskom sveučilišnom studiju**

Osijek, 16.09.2022.

Odboru za završne i diplomske ispite

**Prijedlog ocjene završnog rada na preddiplomskom sveučilišnom studiju**

<b>Ime i prezime Pristupnika:</b>	Filip Hofer
<b>Studij, smjer:</b>	Preddiplomski sveučilišni studij Elektrotehnika i informacijska tehnologija
<b>Mat. br. Pristupnika, godina upisa:</b>	4680, 22.07.2019.
<b>OIB Pristupnika:</b>	39423369785
<b>Mentor:</b>	izv. prof. dr.sc. Josip Job
<b>Sumentor:</b>	,
<b>Sumentor iz tvrtke:</b>	Filip Bajić
<b>Naslov završnog rada:</b>	Klasificiranje grafičkih prikaza korištenjem plitkih neuronskih mreža
<b>Znanstvena grana rada:</b>	<b>Umjetna inteligencija (zn. polje računarstvo)</b>
<b>Zadatak završnog rad:</b>	Zadatak ovog završnog rada je proučiti i opisati plitke neuronske mreže te usporediti njihovu uspješnost u problemu klasificiranja različitih vrsta grafičkih prikaza. Potrebno je pronaći najmanju moguću konvolucijsku neuronsku mrežu sa što manjim brojem slojeva i usporediti je s nekom od najčešće korištenih neuronskih mreža kao što su VGG, ResNet, Inception, AlexNet i dr. Za usporedbu je potrebno koristiti jedinstveni podatkovni skup, npr. ICDAR 2019 dataset of chart images. Tema rezervirana za: Filip Hofer Sumentor iz tvrtke: Filip Bajić (Sveučilišni
<b>Prijedlog ocjene završnog rada:</b>	Izvrstan (5)
<b>Kratko obrazloženje ocjene prema Kriterijima za ocjenjivanje završnih i diplomskih radova:</b>	Primjena znanja stečenih na fakultetu: 3 bod/boda Postignuti rezultati u odnosu na složenost zadatka: 2 bod/boda Jasnoća pismenog izražavanja: 2 bod/boda Razina samostalnosti: 3 razina
<b>Datum prijedloga ocjene od strane mentora:</b>	16.09.2022.
<b>Datum potvrde ocjene od strane Odbora:</b>	21.09.2022.
<b>Potvrda mentora o predaji konačne verzije rada:</b>	<i>Mentor elektronički potpisao predaju konačne verzije.</i>
	Datum: